AMENDMENTS TO THE CLAIMS

1. (Currently amended) A system for detecting an exceeding of time conditions of at least one application executed by a processor, comprising:

a storage element for storing [[the]] a plurality of time conditions associated with rights of use of the at least one application, wherein [[said]] the plurality of time conditions ~~comprise deadlines and~~ are stored as sorted [[by]] in a ~~an increasing deadline~~ chronological order;

a work register for storing a time condition from the plurality of time conditions closest to a current date of the system; and

a comparator for comparing ~~a deadline of~~ the time condition ~~contained~~ stored in the work register with the current date of the system and, [[if]] when the current date of the system exceeds the time condition ~~deadline~~:

providing an interrupt to the processor; and

updating the work register by introducing to the work register a next time condition from the plurality of time conditions stored in the storage element, wherein the next time condition is next closest to the current date.

2. (Previously presented) The system of claim 1, further comprising a timer for calculating the current date of the system, said timer being separate from a counter used by the processor.

3. (Previously presented) The system of claim 1, wherein said storage element contains, with each stored time condition, an identifier of an application with which the time condition is associated.

4. (Previously presented) The system of claim 1, wherein said storage element contains, with each time condition, an identifier of a monitored type of event.

5. (Previously presented) A system for managing rights of use of a digital content linked to at least one time condition, exploiting the detection system of claim 1.

6. (Currently amended) The method of claim 9 comprising, upon each execution of a new application:

inputting a new time condition corresponding to the new application into said storage element so as to maintain storing the time conditions in said storage element in the ~~increasing deadline~~ chronological order; and

updating the work register by introducing to the work register the new time condition if the new time condition is the closest to the current date.

7. (Currently amended) The method of claim 9, comprising, at each stop of the at least one application being executed:

updating the storage element by deleting a time condition corresponding to the at least one application; and

updating the work register if the deleted time condition ~~has a deadline~~ is closest to the current date by placing in the work register a next time condition from the plurality of time conditions stored in the storage element.

8. (Canceled)

9. (Currently amended) A method for detecting an exceeding of time conditions of at least one application executed by a processor, comprising:

storing [[the]] a plurality of time conditions associated with rights of use of at least one application in a storage element, wherein the plurality of time conditions ~~comprise deadlines and~~ are stored as sorted [[by]] in a ~~an increasing deadline~~ chronological order;

storing a time condition from the plurality of time conditions closest to a current date in a work register;

comparing a deadline of the time condition ~~contained~~ stored in the work register with the current date to determine if the current date ~~of the system~~ exceeds the time condition ~~deadline~~; and

[[if]] when it is determined that the current date ~~of the system~~ exceeds the time condition ~~deadline~~:

providing an interrupt to the processor; and

updating the work register by introducing to the work register a next time condition from the plurality of time conditions stored in the storage element, wherein the next time condition is next closest to the current date.

10. (Previously presented) The method of claim 9 further comprising storing, in the storage element, for a time condition, an identifier for an application with which the time condition is associated.